Zeynep Kurtulus
240037253
Assignment1_Part1_Report

Q5. What conclusion if any can be drawn from the weight values? How does sex and BMI affect diabetes disease progression? What are the estimated disease progression values for the below examples? [2 marks]

```
Learned Model Weights: Parameter containing:
tensor([[  1.9413, -11.4535,  26.3020,  16.6372, -10.1897,  -2.1343,  -7.4969,
           8.3693,  22.0618,   2.6037, 153.7365]])
Weight for Sex: -11.453505516052246
Weight for BMI: 26.3019962310791
Being female (Sex = 2) reduces the progression of diabetes compared to being male (Sex = 1).
Higher BMI increases the progression of diabetes.
```

As can be seen from the image generated by the code, the weight for the Sex feature is -11.4535, which is negative. This means that being female indicates a slower progression in the diabetes (we have reached this conclusion because females are denoted with 2 which is a

the model learned weight 1 for the sex category, we know that sex is a categorical variable and female is denoted with 2 and male is denoted with 1. This means that the variable for sex can take one of two values: 1 (for male) and 2 (for female). In a linear regression model, each feature (is multiplied by its corresponding weight (for the sex feature lets assume the weigth is denoted as w_{sex}). The prediction value is: sex value x w_{sex}.

- Prediction for males (sex = 1): 1 x w_{sex}
- Prediction for females (sex = 2): 2 x $w_{sex.}$

A negative weight for sex means that higher numbers (like 2 for females) decrease the predicted progression of diabetes. Since females are encoded as 2, and males as 1, the model predicts a lower progression for females when the weight for sex is negative. If the weight for sex were positive, it would mean that being female (encoded as 2) would increase the progression compared to being male (encoded as 1). Since female is encoded as 2, substituting this value into the model effectively adds -11.45 * 2 to the output when predicting for a female, compared to -11.45 * 1 for a male.This negative weight (-11.45) means that, all else being equal, being female (Sex = 2) decreases the model's prediction for diabetes progression more than being male (Sex = 1). So being female can result in a lower rate of diabetes progression compared to males.

In the model BMI is treated as a numerical feature which means the linear regression model will learn a weight for BMI. If the weight for BMI is positive it means that as the BMI value increases the product w_{BMI} x BMI will also increase and we'll end up having a higher predicted value y which in this case indicates higher disease progression.

The linear regression model has learned a weight of 26.30 for BMI, indicating that higher BMI values are associated with increased diabetes progression. Accordingly, individuals with higher BMI values are expected to have greater progression of diabetes.

Estimated disease progression values can be found below (as an output of the code from the notebook)

```
Predicted disease progression values for the examples:
tensor([[ 43.5994],
        [232.6103]])
```

Q6. Try the code with several learning rates that differ by orders of magnitude, and record the training and test set errors. Describe the theory of how changing the learning rate affects learning. What do you observe in the training error? How about the error on the test set? [3 marks]

```
Learning Rate: 0.001
Final Training MSE: 19769.701171875
Test MSE: 18346.244140625
----------
Learning Rate: 0.01
Final Training MSE: 3351.271240234375
Test MSE: 3425.44384765625
----------
Learning Rate: 0.1
Final Training MSE: 2890.02294921875
Test MSE: 2886.149169921875
----------
Learning Rate: 1
Final Training MSE: nan
Test MSE: nan
----------
Learning Rate: 10
Final Training MSE: nan
Test MSE: nan
----------
   Learning Rate  Final Training MSE      Test MSE
0          0.001        19769.701172  18346.244141
1          0.010         3351.271240   3425.443848
2          0.100         2890.022949   2886.149170
3          1.000                 NaN           NaN
4         10.000                 NaN           NaN
```

Learning rate is an important parameter that determines how much the weights of the model are adjusted with respect to the gradient of the MSE function. Small learning rate indicates that the updates on the weights are too small which makes the training process slow but more secure and stable. On the other hand when we have a high learning rate, the updates on the weights become bigger so the training process becomes faster but it is at the expense of making the model unstable resulting in non-convergence which means the model fails to reach an optimal solution.

We started with a small learning rate of 0.001, the model is updating the weights very slowly. Both the training and test errors are relatively high, indicating that the model hasn't converged well. The model is likely underfitting because it is learning too slowly to reach an optimal solution.

Moving on, increasing the learning rate to 0.01 results in a significant reduction in both training and test errors. This suggests that the model is learning more efficiently compared to the previous learning rate, and the updates are helping it converge faster without overshooting.

With a learning rate of 0.1, the model achieves even lower training and test MSE values, which are very close to each other. This indicates that the model is well-fitted with this learning rate, balancing both training accuracy and generalization. This rate appears optimal among the tested values, as it provides the best generalization on the test set.

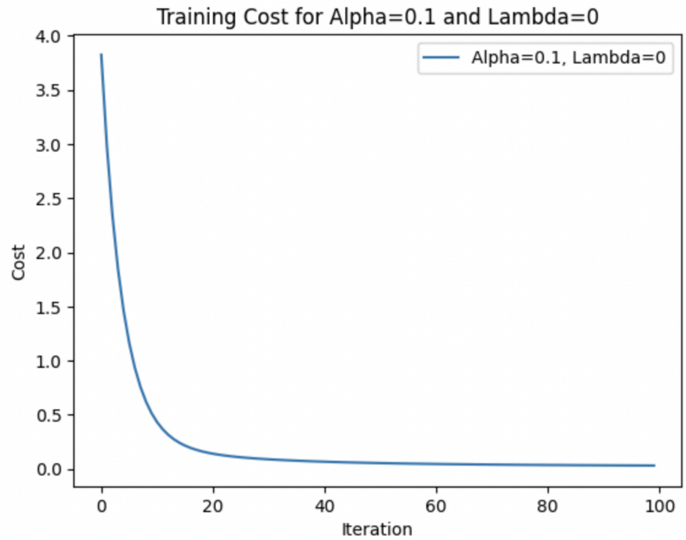
With a high learning rate of 1, both training and test errors become NaN (Not a Number). This indicates that the model's weights are being updated too fast and rapidly that the model fails to converge. The same applies to the learning rate of 10.

So overall, when we consider all different sizes of the learning rates, 0.1 appears to be the ideal one among the others because it minimizes the training and test error without any issues. As the learning rate increases from 0.001 to 0.1, both training and test errors decrease, showing that the model is learning more effectively. When the learning rate becomes too high (1 or 10), the model fails to converge, resulting in NaN errors.

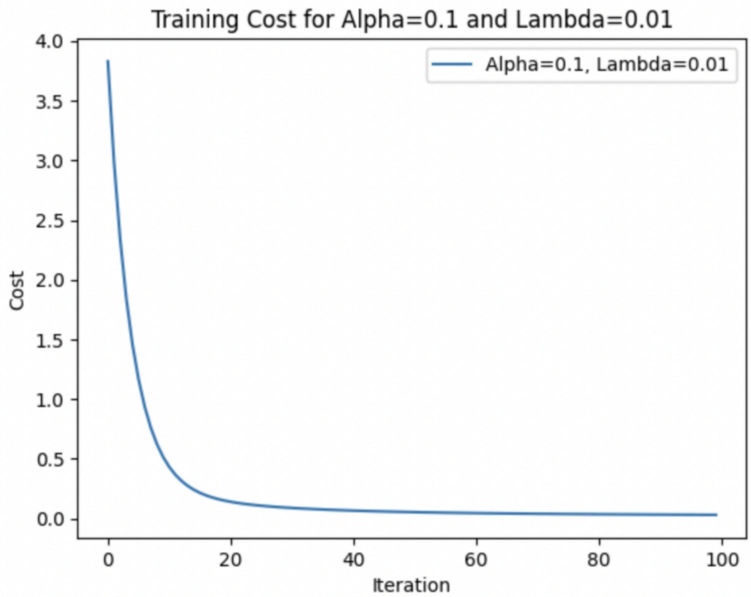
Q8 First of all, find the best value of alpha to use in order to optimize best. Next, experiment with different values of λ and see how this affects the shape of the hypothesis. [3 marks]

```
Alpha: 0.1, Lambda: 10, Final Cost: 0.3052743971347809
Best alpha: 0.1
Best lambda: 0
Minimum cost: 0.031656358391046524
Best model weights: tensor([[ 0.3157, -0.6704,  0.1234, -0.4161,  0.5033, -0.0618]])
```

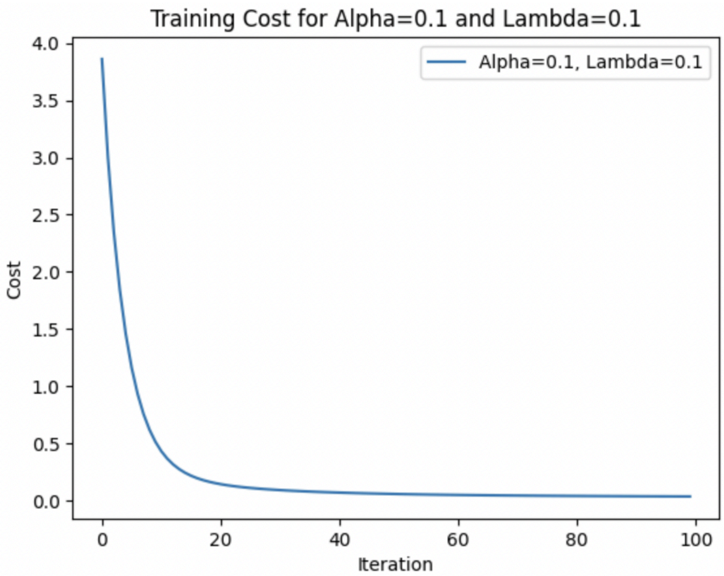
Based on the output which can be seen above the best alpha is 0.1, this is the value that gives the minimum cost when evaluated with different lambda values. In terms of the shape of hypothesis we may look at the below graphs which are the graphs representing different lambda values:



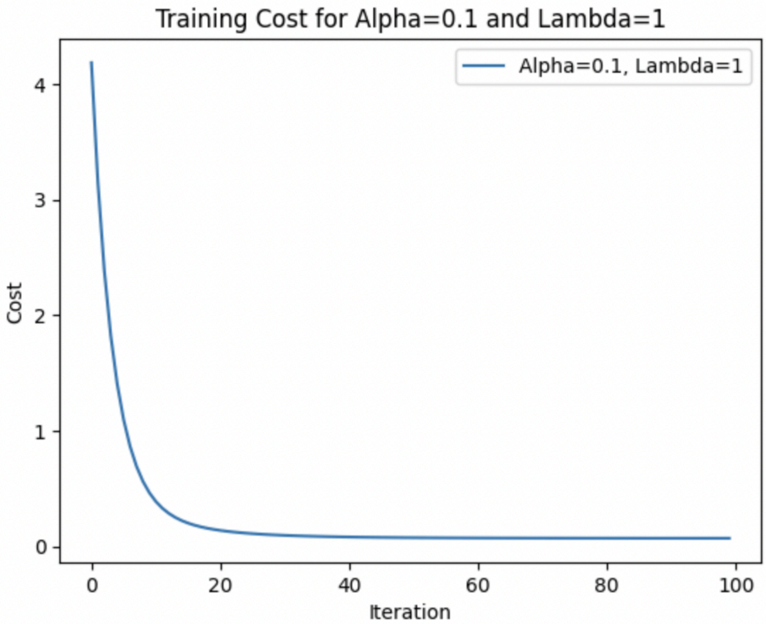

```
Alpha: 0.1, Lambda: 0, Final Cost: 0.031656358391046524
```



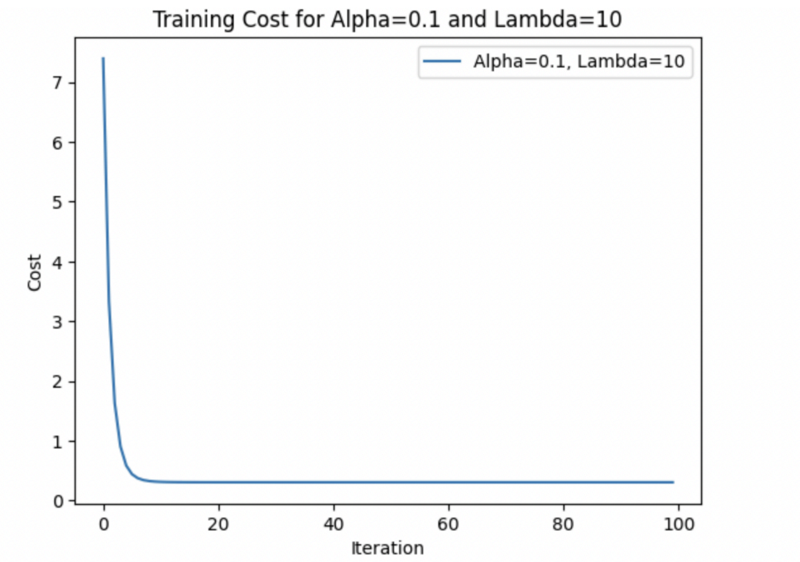

```
Alpha: 0.1, Lambda: 0.01, Final Cost: 0.03185524046421051
```




```
Alpha: 0.1, Lambda: 0.1, Final Cost: 0.03384290635585785
```




```
Alpha: 0.1, Lambda: 1, Final Cost: 0.07071711122989655
```




```
Alpha: 0.1, Lambda: 10, Final Cost: 0.3052743971347809
```

As we have smaller lambda values, we can see that the graph has a sharp decline. The curve shows a steep, exponential-like decline initially, which indicates rapid learning without any regularization constraints. As the lambda values start to get bigger, the curve is still nonlinear but it becomes less steep with less radical declines which indicates the improvements in the regularization. When we have even larger lambda values, the steepness of the curve decreases due to regularization. To sum up when we have small lambda values we see steep, curved line indicating the model is flexible and has a low cost (fits data closely), as the lambda values gets bigger see there is a more linear and gradual decline, the cost increases and the model becomes much less flexible and more constrained.

Zeynep Kurtulus
Student id: 240037253